A Distributed Variational Inference Framework for Unifying Parallel Sparse Gaussian Process Regression Models

Trong Nghia Hoang[§] IDMHTN@NUS.EDU.SG
Quang Minh Hoang[†] HQMINH@COMP.NUS.EDU.SG
Bryan Kian Hsiang Low[†] LOWKH@COMP.NUS.EDU.SG
[†]Department of Computer Science, National University of Singapore, Republic of Singapore
[§]Interactive and Digital Media Institute, National University of Singapore, Republic of Singapore

Abstract

This paper presents a novel distributed variational inference framework that unifies many parallel *sparse Gaussian process regression* (SGPR) models for scalable hyperparameter learning with big data. To achieve this, our framework exploits a structure of correlated noise process model that represents the observation noises as a finite realization of a high-order Gaussian Markov random process. By varying the Markov order and covariance function for the noise process model, different variational SGPR models result. This consequently allows the correlation structure of the noise process model to be characterized for which a particular variational SGPR model is optimal. We empirically evaluate the predictive performance and scalability of the distributed variational SGPR models unified by our framework on two real-world datasets.

1. Introduction

The rich class of Bayesian non-parametric *Gaussian process* (GP) models has recently established itself as a leading approach to probabilistic non-linear regression due to its capability of representing highly complex correlation structure underlying the data. However, the *full-rank GP regression* (FGPR) model incurs a cost of cubic time in the data size for computing the predictive distribution and in each iteration of gradient ascent to refine the estimate of its hyperparameters to improve the log-marginal likelihood (Rasmussen & Williams, 2006), hence limiting its usage to only small datasets in practice.

To boost its scalability, a wealth of *sparse GPR* (SGPR) models (Lázaro-Gredilla et al., 2010; Low et al., 2015; Quiñonero-Candela & Rasmussen, 2005; Snelson & Ghahramani, 2007) utilizing varying low-rank approximate representations of FGPR have been proposed, many of which are spanned by the unifying view of Quiñonero-Candela & Rasmussen (2005) based on the notion of inducing variables (Section 3) such as the *subset of regressors* (SoR) (Smola & Bartlett, 2001), *deterministic training conditional* (DTC) (Seeger et al., 2003), *fully independent training conditional* (FITC) (Snelson & Gharahmani, 2005), *fully independent conditional* (FIC), *partially independent training conditional* (PITC) (Schwaighofer & Tresp, 2003), and *partially independent conditional* (PIC) (Snelson & Ghahramani, 2007) approximations. Consequently, they incur linear time in the data size, which is still prohibitively expensive for training with million-sized datasets. To scale up these SGPR models further for performing real-time predictions necessary in many time-critical applications and decision support systems (e.g., environmental sensing (Cao et al., 2013; Dolan et al., 2009; Ling et al., 2016; Low et al., 2008; 2009; 2011; 2012; Podnar et al., 2010; Zhang et al., 2016), traffic monitoring (Chen et al., 2012; 2013b; 2015; Hoang et al., 2014a;b; Low et al., 2014a;b; Ouyang et al., 2014; Xu et al., 2014; Yu et al., 2012)), a number of these models have been parallelized (e.g., FITC, FIC, PITC, and PIC (Chen et al., 2013a), and *low-rank-cum-Markov approximation* (LMA) (Low et al., 2015) unifying a spectrum of SGPR models with PIC and FGPR at the two extremes), but the resulting parallel SGPR models do not readily extend to include hyperparameter learning. The work of Deisenroth & Ng (2015) has recently introduced a practical product-of-expert (PoE) paradigm for GP which imposes a factorized structure on the marginal likelihood that allows it to be optimized effectively in a parallel/distributed fashion.

However, the main criticism of the above approximation paradigms is their lack of a rigorous approximation since they do not require optimizing some loss criterion incurred by an approximation model (Titsias, 2009b). To resolve this, the work of Titsias (2009a) has introduced an alterna-

Proceedings of the 33^{rd} International Conference on Machine Learning, New York, NY, USA, 2016. JMLR: W&CP volume 48.

tive formulation of variational inference for DTC approximation that involves minimizing the *Kullback-Leibler* (KL) distance between the variational DTC approximation and the posterior distribution of some latent variables (including the inducing variables) induced by the FGPR model given the data/observations, or equivalently maximizing a lower bound of the log-marginal likelihood. Hyperparameter learning can then be achieved by maximizing this variational lower bound as a function of the hyperparameters. Its incurred time per iteration of gradient ascent is still linear in the data size but can be significantly reduced by parallelization on multiple distributed machines/cores, as demonstrated by Gal et al. (2014). Despite their theoretical rigor and scalability, it has been shown by Hoang et al. (2015) that DTC has utilized the most crude approximation among all SGPR models (except SoR) spanned by the unifying view of Quiñonero-Candela & Rasmussen (2005), thus severely compromising its predictive performance. As such, it remains an open question whether efficient and scalable hyperparameter learning of more refined SGPR models (e.g., PIC, LMA) for big data can be achieved through distributed variational inference[1].

To address this question, we first observe that variational DTC (Titsias, 2009a) and its distributed variant (Gal et al., 2014) have implicitly assumed the observation noises to be independently distributed with constant variance, which is often violated in practice (Huizenga & Molenaar, 1995; Koochakzadeh et al., 2015; Rasmussen & Williams, 2006). This strong assumption has been relaxed slightly by Titsias (2009b) to that of input-dependent noise variance which allows a variational inference formulation for FITC approximation to be derived. This seems to suggest a possibility of deriving variational inference formulations for the more refined sparse GP approximations and, perhaps surprisingly, their distributed variants by exploiting more sophisticated noise process models such as those being used by existing GP works. Such GP works, however, suffer from poor scalability to big data: Notably, the work of Goldberg et al. (1997) has proposed a *heteroscedastic GPR* (HGPR) model that extends the FGPR model by representing the noise variance with a log-GP (in addition to the original GP modeling the noise-free latent measurements), hence allowing it to vary across the input space; the observation noises remain independently distributed though. But, the exact HGPR model cannot be computed tractably while approximate HGPR models (Kersting et al., 2007; Lázaro-Gredilla & Titsias, 2011) still incur cubic time in the data size, thus scaling poorly to big data (i.e., million-sized datasets). This is similarly true for FGPR models (Murray-Smith & Girard, 2001; Rasmussen & Williams, 2006) that represent correlation of observation noises with an additional covariance function. Unfortunately, variational DTC (Titsias, 2009a), its distributed variant (Gal et al., 2014), and variational FITC (Titsias, 2009b) cannot readily accommodate such heteroscedastic or correlated noise process models without sacrificing their time efficiency. So, the key challenge remains in being able to specify some structure of the noise process model that can be exploited for efficient and scalable hyperparameter learning of more refined SGPR models (e.g., PITC, PIC, LMA) through distributed variational inference, which is the focus of our work here.

To tackle this challenge, this paper presents a novel variational inference framework (Section 3) for deriving sparse GP approximations to a new FGPR model with observation noises that vary as a finite realization of a high-order Gaussian Markov random process (Section 2), thus enriching the expressiveness of HGPR models by correlating the noises across the input space. Interestingly, our proposed framework can unify many SGPR models via specific choices of the Markov order and covariance function for the noise process model (Section 4), which include variational DTC and FITC (Titsias, 2009a;b) and the more refined PITC, PIC, and LMA. This then enables the characterization of the correlation structure of the noise process model for which a particular sparse GP approximation is (variationally) optimal[2] and explains why PIC and LMA tend to outperform DTC and FITC in practice despite not being characterized as optimal when independently distributed observation noises are assumed. More importantly, our framework is amenable to parallelization by distributing its computational load of hyperparameter learning on multiple machines/cores (Section 5), hence reducing its incurred linear time per iteration of gradient ascent by a factor close to the number of machines/cores. We empirically evaluate the predictive performance and scalability of the distributed variational SGPR models (e.g., state-of-the-art distributed variational DTC (Gal et al., 2014)) unified by our framework on two real-world datasets (Section 6).

2. Gaussian Processes with Correlated Noises

Let $\mathcal{X}$ be a set representing the input domain such that each d-dimensional input feature vector $\mathbf{x} \in \mathcal{X}$ is associated with a latent output variable $f_{\mathbf{x}}$ and its corresponding noisy output $y_{\mathbf{x}} \triangleq f_{\mathbf{x}} + \epsilon_{\mathbf{x}}$ differing by an additive noise $\epsilon_{\mathbf{x}}$. Let $\{f_{\mathbf{x}}\}_{\mathbf{x}\in\mathcal{X}}$ denote a *Gaussian process* (GP), that is, every fi-

[1] The work of Campbell et al. (2015) has separately developed a distributed variational inference framework for Bayesian nonparametric models that are limited to only clustering processes (e.g., Dirichlet, Pitman-Yor, and their variants) not including GPs.

[2] Such a characterization, which is important to many real-world applications of GP involving different noise structures, cannot be realized from the unifying framework of Hoang et al. (2015) relying on reverse variational inference to obtain the variational lower bound for a SGPR model. Furthermore, it is unclear or at least non-trivial to determine whether it is amenable to parallelization for learning the hyperparameters of LMA which does not meet its assumed decomposability conditions.

nite subset of $\{f_{\mathbf{x}}\}_{\mathbf{x}\in\mathcal{X}}$ follows a multivariate Gaussian distribution. Then, the GP is fully specified by its *prior* mean $\mu_{\mathbf{x}} \triangleq \mathbb{E}[f_{\mathbf{x}}]$ and covariance $k_{\mathbf{x}\mathbf{x}'} \triangleq \mathrm{cov}[f_{\mathbf{x}}, f_{\mathbf{x}'}]$ for all $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$, the latter of which can be defined, for example, by the widely-used squared exponential covariance function $k_{\mathbf{x}\mathbf{x}'} \triangleq \sigma_s^2 \exp(-0.5(\mathbf{x}-\mathbf{x}')^\top \mathbf{\Lambda}^{-2}(\mathbf{x}-\mathbf{x}'))$ where $\mathbf{\Lambda} \triangleq \mathrm{diag}[\ell_1, \ldots, \ell_d]$ and σ_s^2 are its length-scale and signal variance hyperparameters, respectively. Similarly, let $\{\epsilon_{\mathbf{x}}\}_{\mathbf{x}\in\mathcal{X}}$ denote another GP with *prior* mean $\mathbb{E}[\epsilon_{\mathbf{x}}] = 0$ and covariance $k^{\epsilon}_{\mathbf{x}\mathbf{x}'} \triangleq \mathrm{cov}[\epsilon_{\mathbf{x}}, \epsilon_{\mathbf{x}'}]$ for all $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$, the latter of which is defined by a covariance function like that used for $k_{\mathbf{x}\mathbf{x}'}$ (albeit with different hyperparameter values).

Supposing a column vector $y_{\mathcal{D}} \triangleq (y_{\mathbf{x}})^\top_{\mathbf{x}\in\mathcal{D}}$ of noisy outputs is observed for some set $\mathcal{D} \subset \mathcal{X}$ of training inputs, a FGPR model with correlated observation noises (Murray-Smith & Girard, 2001; Rasmussen & Williams, 2006) can perform probabilistic regression by providing a GP *posterior*/predictive distribution $p(f_{\mathcal{U}}|y_{\mathcal{D}}) = \mathcal{N}(f_{\mathcal{U}}|\mu_{\mathcal{U}} + K_{\mathcal{U}\mathcal{D}}(K_{\mathcal{D}\mathcal{D}} + S_{\mathcal{D}\mathcal{D}})^{-1}(y_{\mathcal{D}} - \mu_{\mathcal{D}}), K_{\mathcal{U}\mathcal{U}} - K_{\mathcal{U}\mathcal{D}}(K_{\mathcal{D}\mathcal{D}} + S_{\mathcal{D}\mathcal{D}})^{-1}K_{\mathcal{D}\mathcal{U}})$ of the unobserved outputs $f_{\mathcal{U}} \triangleq (f_{\mathbf{x}})^\top_{\mathbf{x}\in\mathcal{U}}$ for any set $\mathcal{U} \subset \mathcal{X} \setminus \mathcal{D}$ of test inputs where $\mu_{\mathcal{U}}$ ($\mu_{\mathcal{D}}$) is a column vector with mean components $\mu_{\mathbf{x}}$ for all $\mathbf{x} \in \mathcal{U}$ ($\mathbf{x} \in \mathcal{D}$), $K_{\mathcal{U}\mathcal{D}}$ ($K_{\mathcal{D}\mathcal{D}}$) is a matrix with covariance components $k_{\mathbf{x}\mathbf{x}'}$ for all $\mathbf{x} \in \mathcal{U}, \mathbf{x}' \in \mathcal{D}$ ($\mathbf{x}, \mathbf{x}' \in \mathcal{D}$), $K_{\mathcal{D}\mathcal{U}} = K_{\mathcal{U}\mathcal{D}}^\top$, and $S_{\mathcal{D}\mathcal{D}}$ is a matrix with covariance components $k^{\epsilon}_{\mathbf{x}\mathbf{x}'}$ for all $\mathbf{x}, \mathbf{x}' \in \mathcal{D}$ representing the correlation of observation noises $\epsilon_{\mathcal{D}} \triangleq (\epsilon_{\mathbf{x}})^\top_{\mathbf{x}\in\mathcal{D}} \sim \mathcal{N}(0, S_{\mathcal{D}\mathcal{D}})$ which implies $p(y_{\mathcal{D}}|f_{\mathcal{D}}) = \mathcal{N}(y_{\mathcal{D}}|f_{\mathcal{D}}, S_{\mathcal{D}\mathcal{D}})$. However, the FGPR model scales poorly in the size $|\mathcal{D}|$ of data because computing the GP predictive distribution incurs $\mathcal{O}(|\mathcal{D}|^3)$ time due to the inversion of $K_{\mathcal{D}\mathcal{D}} + S_{\mathcal{D}\mathcal{D}}$.

To improve its scalability, our key idea stems from imposing a B-th order Markov property on the observation noise process: Specifically, let the set $\mathcal{D}$ ($\mathcal{U}$) of training (test) inputs be partitioned evenly into M disjoint subsets $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_M$ ($\mathcal{U}_1, \mathcal{U}_2, \ldots, \mathcal{U}_M$). In the same spirit as a Gaussian Markov random process, imposing a B-th order Markov property on the observation noise process $\{\epsilon_{\mathbf{x}}\}_{\mathbf{x}\in\mathcal{D}}$ with respect to such a partition implies that observation noises $\epsilon_{\mathcal{D}_i}$ and $\epsilon_{\mathcal{D}_j}$ are conditionally independent given $\epsilon_{\mathcal{D}\setminus(\mathcal{D}_i\cup\mathcal{D}_j)}$ if $|i-j| > B$. As shown in (Low et al., 2015), this can be realized by partitioning the matrix $S_{\mathcal{V}\mathcal{V}}$ for the entire set $\mathcal{V} \triangleq \mathcal{D}\cup\mathcal{U}$ of training and test inputs into $M \times M$ square blocks, that is, $S_{\mathcal{V}\mathcal{V}} \triangleq [S_{\mathcal{V}_i\mathcal{V}_j}]_{i,j=1,\ldots,M}$ where $\mathcal{V}_i \triangleq \mathcal{D}_i \cup \mathcal{U}_i$ (hence, $\mathcal{V} = \bigcup_{i=1}^M \mathcal{V}_i$) and

$$S_{\mathcal{V}_i\mathcal{V}_j} \triangleq \begin{cases} K^{\epsilon}_{\mathcal{V}_i\mathcal{V}_j} & \text{if } |i-j| \leq B, \\ K^{\epsilon}_{\mathcal{V}_i\mathcal{D}_i^B} {K^{\epsilon}_{\mathcal{D}_i^B\mathcal{D}_i^B}}^{-1} S_{\mathcal{D}_i^B\mathcal{V}_j} & \text{if } j-i > B > 0, \\ S_{\mathcal{V}_i\mathcal{D}_j^B} {K^{\epsilon}_{\mathcal{D}_j^B\mathcal{D}_j^B}}^{-1} K^{\epsilon}_{\mathcal{D}_j^B\mathcal{V}_j} & \text{if } i-j > B > 0, \\ \underline{0} & \text{if } |i-j| > B = 0; \end{cases} \quad (1)$$

such that $K^{\epsilon}_{\mathcal{V}_i\mathcal{V}_j} \triangleq [k^{\epsilon}_{\mathbf{x}\mathbf{x}'}]_{\mathbf{x}\in\mathcal{V}_i,\mathbf{x}'\in\mathcal{V}_j}$, $\mathcal{D}_i^B = \mathcal{D}_{i+1} \cup \ldots \cup \mathcal{D}_{i_B^+}$ where $i_B^+ \triangleq \min(M, i+B)$, and $\underline{0}$ denotes a square block comprising components of value 0. Though the matrix $S_{\mathcal{V}\mathcal{V}}$ (and hence its submatrix $S_{\mathcal{D}\mathcal{D}}$) is dense, its structure (1) interestingly guarantees that $P_{\mathcal{D}\mathcal{D}} \triangleq S_{\mathcal{D}\mathcal{D}}^{-1}$ is B-block-banded (Low et al., 2015). That is, if $|i-j| > B$, then $P_{\mathcal{D}_i\mathcal{D}_j} = \underline{0}$. Such sparsity of $P_{\mathcal{D}\mathcal{D}}$ is the main ingredient to improving the scalability of the FGPR model with correlated observation noises, as revealed in Section 3.

3. Variational Sparse GP Regression Models

This section introduces a novel variational inference framework for deriving sparse GP approximations to the FGPR model with correlated observation noises that vary as a finite realization of a B-th order Gaussian Markov random process (Section 2). Similar to the variational inference formulations for DTC and FITC approximations (Titsias, 2009a;b), we exploit a vector $f_{\mathcal{S}}$ of inducing output variables for some small set $\mathcal{S} \subset \mathcal{X}$ of inducing inputs to construct sufficient statistics for $y_{\mathcal{D}}$ such that $y_{\mathcal{D}} \perp f_{\mathcal{D}} \mid f_{\mathcal{S}}$ (i.e., $p(f_{\mathcal{D}}, f_{\mathcal{S}}|y_{\mathcal{D}}) = p(f_{\mathcal{D}}|f_{\mathcal{S}})\ p(f_{\mathcal{S}}|y_{\mathcal{D}})$). However, choosing $\mathcal{S}$ for which this conditional independence property holds may not be possible in practice. Instead, it is approximated by some (heuristic) choice of $\mathcal{S}$ as follows:

$$p(f_{\mathcal{D}}, f_{\mathcal{S}}|y_{\mathcal{D}}) \simeq p(f_{\mathcal{D}}|f_{\mathcal{S}})\ q(f_{\mathcal{S}}) \quad (2)$$

where $q(f_{\mathcal{S}})$ is a free-form distribution to be optimized by minimizing the KL distance $D_{\mathrm{KL}}(q) \triangleq \mathrm{KL}(p(f_{\mathcal{D}}|f_{\mathcal{S}})q(f_{\mathcal{S}})||p(f_{\mathcal{D}}, f_{\mathcal{S}}|y_{\mathcal{D}}))$ between $p(f_{\mathcal{D}}|f_{\mathcal{S}})\ q(f_{\mathcal{S}})$ and $p(f_{\mathcal{D}}, f_{\mathcal{S}}|y_{\mathcal{D}})$ on either sides of (2) with respect to $q(f_{\mathcal{S}})$ and the hyperparameters defining prior covariances $k_{\mathbf{x}\mathbf{x}'}$ and $k^{\epsilon}_{\mathbf{x}\mathbf{x}'}$ (Section 2). To do this, note that

$$\log p(y_{\mathcal{D}}) = \mathcal{L}(q, \mathcal{Z}) + D_{\mathrm{KL}}(q) \quad (3)$$

where $\mathcal{Z}$ denotes a set of hyperparameters [3] defining $k_{\mathbf{x}\mathbf{x}'}$ and $k^{\epsilon}_{\mathbf{x}\mathbf{x}'}$ and, as derived in Appendix A,

$$\begin{aligned}\mathcal{L}(q, \mathcal{Z}) \quad &\triangleq \mathbb{E}_{q(f_{\mathcal{S}})}[\log \mathcal{N}(y_{\mathcal{D}}|\nu, S_{\mathcal{D}\mathcal{D}}) + \log(p(f_{\mathcal{S}})/q(f_{\mathcal{S}}))] \\ &\quad - 0.5\mathrm{Tr}[S_{\mathcal{D}\mathcal{D}}^{-1}(K_{\mathcal{D}\mathcal{D}} - Q_{\mathcal{D}\mathcal{D}})]\end{aligned} \quad (4)$$

where $\nu \triangleq \mu_{\mathcal{D}} + K_{\mathcal{D}\mathcal{S}}K_{\mathcal{S}\mathcal{S}}^{-1}(f_{\mathcal{S}} - \mu_{\mathcal{S}})$ and $Q_{\mathcal{Y}\mathcal{Y}'} \triangleq K_{\mathcal{Y}\mathcal{S}}K_{\mathcal{S}\mathcal{S}}^{-1}K_{\mathcal{S}\mathcal{Y}'}$ for all $\mathcal{Y}, \mathcal{Y}' \subset \mathcal{X}$. Supposing $\mathcal{Z}$ is fixed/known, since $D_{\mathrm{KL}}(q) \geq 0$ and $\log p(y_{\mathcal{D}})$ is independent of $q(f_{\mathcal{S}})$, minimizing $D_{\mathrm{KL}}(q)$ is equivalent to maximizing $\mathcal{L}(q, \mathcal{Z})$ which entails solving for the optimal choice of $q(f_{\mathcal{S}})$ that satisfies $\partial\mathcal{L}(q, \mathcal{Z})/\partial q = 0$ as a PDF parameterized by $\mathcal{Z}$. As derived in Appendix B, this yields

$$\log q(f_{\mathcal{S}}) = \log[\mathcal{N}(y_{\mathcal{D}}|\nu, S_{\mathcal{D}\mathcal{D}})\ p(f_{\mathcal{S}})] + \mathrm{const} \quad (5)$$

where const is used to absorb all terms independent of $f_{\mathcal{S}}$. By completing the quadratic form for $f_{\mathcal{S}}$ (Appendix C), it can be derived from (5) that

$$q(f_{\mathcal{S}}) = \mathcal{N}(f_{\mathcal{S}}|\mu_{\mathcal{S}} + K_{\mathcal{S}\mathcal{S}}\Gamma_{\mathcal{S}\mathcal{S}}^{-1}V_{\mathcal{S}\mathcal{D}}, K_{\mathcal{S}\mathcal{S}}\Gamma_{\mathcal{S}\mathcal{S}}^{-1}K_{\mathcal{S}\mathcal{S}}) \quad (6)$$

[3] All the terms in (3) and the equations to follow depend on $\mathcal{Z}$. To ease notational clutter, their dependence on $\mathcal{Z}$ are omitted from their expressions, unless otherwise needed for clarity reasons.

where $\Gamma_{\mathcal{SS}} \triangleq K_{\mathcal{SS}} + K_{\mathcal{SD}}P_{\mathcal{DD}}K_{\mathcal{DS}}$, $P_{\mathcal{DD}} = S^{-1}_{\mathcal{DD}}$, and $V_{\mathcal{SD}} \triangleq K_{\mathcal{SD}}P_{\mathcal{DD}}(y_{\mathcal{D}} - \mu_{\mathcal{D}})$. By plugging (6) into (4), $\mathcal{R}(\mathcal{Z}) \triangleq \max_q \mathcal{L}(q, \mathcal{Z})$ can be derived analytically. Then, maximizing $\mathcal{R}(\mathcal{Z})$ with respect to $\mathcal{Z}$ gives the optimal hyperparameters[4]. The details of this maximization are deferred to Section 5 for the sake of clarity. In the rest of this section, we will instead focus on deriving an equivalent reformulation of $q(f_{\mathcal{S}})$ (6) that can be computed efficiently in linear time in the data size $|\mathcal{D}|$ since computing $q(f_{\mathcal{S}})$ directly using (6) generally incurs $\mathcal{O}(|\mathcal{D}|^3)$ time. Moreover, its computation can be distributed among parallel machines/cores to achieve scalability, as described next.

Our first result (Lemma 1) below derives the structure of each *non-zero* constituent block $P_{\mathcal{D}_i\mathcal{D}_j}$ of the B-block banded matrix $P_{\mathcal{DD}}$ (i.e., $|i-j| \leq B$), which is the key ingredient to be exploited for computing $q(f_{\mathcal{S}})$ (6) efficiently and scalably (Theorem 1). Note that it suffices to simply compute $P_{\mathcal{D}_i\mathcal{D}_j}$ for $i \leq j \leq i^+_B$ because $P_{\mathcal{DD}}$ is symmetric (i.e., $P_{\mathcal{D}_i\mathcal{D}_j} = P^\top_{\mathcal{D}_j\mathcal{D}_i}$ for $j \leq i \leq j^+_B$) and B-block-banded (i.e., $P_{\mathcal{D}_i\mathcal{D}_j} = \underline{0}$ for $|i-j| > B$) (Section 2):

Lemma 1 *Let*[5]

$$G^k \triangleq \begin{bmatrix} G^k_{\mathcal{D}_k\mathcal{D}_k} & G^k_{\mathcal{D}_k\mathcal{D}^B_k} \\ G^k_{\mathcal{D}^B_k\mathcal{D}_k} & G^k_{\mathcal{D}^B_k\mathcal{D}^B_k} \end{bmatrix} \triangleq \begin{bmatrix} S_{\mathcal{D}_k\mathcal{D}_k} & S_{\mathcal{D}_k\mathcal{D}^B_k} \\ S_{\mathcal{D}^B_k\mathcal{D}_k} & S_{\mathcal{D}^B_k\mathcal{D}^B_k} \end{bmatrix}^{-1}$$

for $k = 1, \ldots, M$. *Then, for* $i, j = 1, \ldots, M$ *such that* $i \leq j \leq i^+_B = \min(M, i+B)$ *and* $j^-_B \triangleq \max(1, j-B)$,

$$P_{\mathcal{D}_i\mathcal{D}_j} = \textstyle\sum_{k=j^-_B}^{i} G^k_{\mathcal{D}_i\mathcal{D}_k} G^{k^{-1}}_{\mathcal{D}_k\mathcal{D}_k} G^k_{\mathcal{D}_k\mathcal{D}_j} \; . \quad (7)$$

See Appendix D for its proof. From Lemma 1, constructing G^k of size $\mathcal{O}(B|\mathcal{D}|/M) = \mathcal{O}(B|\mathcal{S}|)$[6] by $\mathcal{O}(B|\mathcal{S}|)$ incurs $\mathcal{O}(B^3|\mathcal{S}|^3)$ time. $G^1, \ldots, G^M$ can be computed independently and hence in parallel on M distributed machines/cores; otherwise, they can be constructed sequentially in $\mathcal{O}(|\mathcal{D}||\mathcal{S}|^2B^3) = \mathcal{O}(MB^3|\mathcal{S}|^3)$ time.

Our next result exploits Lemma 1 and the B-block-banded structure of $P_{\mathcal{DD}}$ for decomposing $\Gamma_{\mathcal{SS}}$ and $V_{\mathcal{SD}}$ in (6) into linear sums of independent terms whose computations can therefore be distributed among parallel machines/cores:

Theorem 1 *Let* $\mathcal{B}^+(k) \triangleq \{k, k+1, \ldots, k^+_B\}$. *Then,* $\Gamma_{\mathcal{SS}} = K_{\mathcal{SS}} + \sum_{k=1}^M \beta_k$ *and* $V_{\mathcal{SD}} = \sum_{k=1}^M \alpha_k$ *where*

$$\alpha_k \triangleq \textstyle\sum_{i,j \in \mathcal{B}^+(k)} K_{\mathcal{SD}_i} G^k_{\mathcal{D}_i\mathcal{D}_k} G^{k^{-1}}_{\mathcal{D}_k\mathcal{D}_k} G^k_{\mathcal{D}_k\mathcal{D}_j} (y_{\mathcal{D}_j} - \mu_{\mathcal{D}_j})$$
$$\beta_k \triangleq \textstyle\sum_{i,j \in \mathcal{B}^+(k)} K_{\mathcal{SD}_i} G^k_{\mathcal{D}_i\mathcal{D}_k} G^{k^{-1}}_{\mathcal{D}_k\mathcal{D}_k} G^k_{\mathcal{D}_k\mathcal{D}_j} K_{\mathcal{D}_j\mathcal{S}} \; . \quad (8)$$

Its proof is in Appendix E. From (8), $\alpha_1, \ldots, \alpha_M$ and $\beta_1, \ldots, \beta_M$ can again be computed independently and hence in parallel in $\mathcal{O}(|\mathcal{B}^+(k)|^2|\mathcal{S}|^3) = \mathcal{O}(B^2|\mathcal{S}|^3)$ time on M distributed machines/cores; alternatively, they can be computed sequentially in $\mathcal{O}(|\mathcal{D}||\mathcal{S}|^2B^2) = \mathcal{O}(MB^2|\mathcal{S}|^3)$ time. $\Gamma_{\mathcal{SD}}$ and $V_{\mathcal{SD}}$ can then be constructed in $\mathcal{O}(M|\mathcal{S}|^2)$ time and computing $q(f_{\mathcal{S}})$ (6) in turn incurs $\mathcal{O}(|\mathcal{S}|^3)$ time. So, the overall time complexity of deriving $q(f_{\mathcal{S}})$ (6) is linear in $|\mathcal{D}|$ which can be further reduced via parallelization by a factor close to the number of machines/cores.

Remark The efficiency in deriving $q(f_{\mathcal{S}})$ (6) can be exploited for constructing a linear-time approximation $q(f_{\mathcal{U}_i}|y_{\mathcal{D}})$ to the GP predictive distribution $p(f_{\mathcal{U}_i}|y_{\mathcal{D}})$ for any subset $\mathcal{U}_i = \mathcal{V}_i \setminus \mathcal{D}_i$ of test inputs and its distributed variant, the details of which are not needed to understand our distributed variational inference framework for hyperparameter learning in Section 5 (but required for predictions in our experiments in Section 6) and hence deferred to Appendices F and G, respectively. By varying the choices of the Markov order B, the covariance function $k^\epsilon_{\mathbf{xx}'}$ for the noise process model, the number M of data partitions, and the approximation $q(f_{\mathcal{U}_i}|f_{\mathcal{S}}, y_{\mathcal{D}})$ to the test conditional $p(f_{\mathcal{U}_i}|f_{\mathcal{S}}, y_{\mathcal{D}})$, they recover the predictive distribution of various SGPR models spanned by the unifying view of Quiñonero-Candela & Rasmussen (2005) (i.e., SoR, DTC, FITC, FIC, PITC, PIC) and LMA, as detailed in Section 4.

4. Unifying Sparse GP Regression Models

4.1. SGPR models spanned by unifying view of Quiñonero-Candela & Rasmussen (2005)

We will first demonstrate how our variational inference framework (Section 3) can unify SoR, DTC, FITC, FIC, PITC, PIC. As discussed in Appendix F, it suffices to show how the covariance function $k^\epsilon_{\mathbf{xx}'}$ for the noise process model and the Markov order B can be set such that the resulting variationally optimal distribution $q(f_{\mathcal{S}})$ (6) coincides with that of these SGPR models. Let us consider the following covariance function for the noise process model:

$$k^\epsilon_{\mathbf{xx}'} \triangleq k_{\mathbf{xx}'} - K_{\mathbf{x}\mathcal{S}}K^{-1}_{\mathcal{SS}}K_{\mathcal{S}\mathbf{x}'} + \sigma_n^2\mathbb{I}(\mathbf{x} = \mathbf{x}') \; . \quad (9)$$

where σ_n^2 is a noise variance hyperparameter. It follows that $K^\epsilon_{\mathcal{DD}} = K_{\mathcal{DD}} - K_{\mathcal{DS}}K^{-1}_{\mathcal{SS}}K_{\mathcal{SD}} + \sigma_n^2 I = K_{\mathcal{DD}} - Q_{\mathcal{DD}} + \sigma_n^2 I$. Then, imposing the B-th order Markov property on the observation noise process $\{\epsilon_{\mathbf{x}}\}_{\mathbf{x}\in\mathcal{D}}$ through (1) with $B = 0$ gives $S_{\mathcal{DD}} = \text{blkdiag}[K_{\mathcal{DD}} - Q_{\mathcal{DD}}] + \sigma_n^2 I$ which implies $S_{\mathcal{D}_i\mathcal{D}_j} = \mathbb{I}(i=j)(K_{\mathcal{D}_i\mathcal{D}_i} - Q_{\mathcal{D}_i\mathcal{D}_i} + \sigma_n^2 I)$. Plugging this expression of $S_{\mathcal{DD}}$ into (6) yields the same $q(f_{\mathcal{S}})$ induced by PIC, as detailed in (Hoang et al., 2015)[7].

[4] The formulation described thus far is reminiscent of variational *expectation-maximization* (EM) (Wainwright & Jordan, 2008) whose *approximate* E step involves maximizing the variational lower bound $\mathcal{L}(q, \mathcal{Z})$ with respect to q given optimal $\mathcal{Z}$ from M step while its M step follows exactly that of EM by maximizing $\mathcal{L}(q, \mathcal{Z})$ with respect to $\mathcal{Z}$ given optimal q from E step. The implication is that, like variational EM, variational DTC and FITC (Titsias, 2009a;b), and distributed variational DTC (Gal et al., 2014), though the log-marginal likelihood $\log p(y_{\mathcal{D}})$ does not necessarily increase in each iteration, our formulation has an attractive interpretation of maximizing its lower bound.

[5] When $B = 0$, $G^k = S^{-1}_{\mathcal{D}_k\mathcal{D}_k}$ for $k = 1, \ldots, M$.

[6] We choose the size of $\mathcal{S}$ such that $|\mathcal{D}_i| = |\mathcal{D}|/M = \mathcal{O}(|\mathcal{S}|)$.

[7] The work of Hoang et al. (2015) assumes a GP with zero prior mean which is equivalent to setting $\mu_{\mathcal{S}} = \mu_{\mathcal{D}} = \underline{0}$ in (6).

Furthermore, if $|\mathcal{D}_i| = 1$ for $i = 1, \ldots, M$ (i.e., $M = |\mathcal{D}|$), then $S_{\mathcal{D}\mathcal{D}} = \mathrm{diag}[K_{\mathcal{D}\mathcal{D}} - Q_{\mathcal{D}\mathcal{D}}] + \sigma_n^2 I$ which can be plugged into (6) to recover the same $q(f_{\mathcal{S}})$ induced by FIC. Supposing $k^{\epsilon}_{\mathbf{x}\mathbf{x}'} \triangleq \sigma_n^2 \mathbb{I}(\mathbf{x} = \mathbf{x}')$ instead of using (9), $S_{\mathcal{D}\mathcal{D}} = \sigma_n^2 I$ that can be plugged into (6) to recover the same $q(f_{\mathcal{S}})$ induced by DTC. Finally, the predictive distributions $q(f_{\mathcal{U}_i}|y_{\mathcal{D}})$ of the SGPR models spanned by the unifying view of Quiñonero-Candela & Rasmussen (2005) such as PIC, FIC and DTC can be recovered by integrating the resulting $q(f_{\mathcal{S}})$ with their approximations $q(f_{\mathcal{U}_i}|f_{\mathcal{S}}, y_{\mathcal{D}})$ to the test conditional $p(f_{\mathcal{U}_i}|f_{\mathcal{S}}, y_{\mathcal{D}})$, as discussed in Appendix F. We omit details of unifying PITC, FITC, and SoR with our framework since they induce the same $q(f_{\mathcal{S}})$ as that of PIC, FIC and DTC, respectively.

4.2. Low-rank-cum-Markov approximation (LMA)

To unify LMA with our variational inference framework, we have to derive its induced $q(f_{\mathcal{S}})$ since it is not explicitly given in (Low et al., 2015). To do this, let us first derive an equivalent reformulation of the predictive distribution of the FGPR model with independently distributed observation noises of constant variance, as shown in Appendix H: $p(f_{\mathcal{S}}|y_{\mathcal{D}}) \triangleq \mathcal{N}(f_{\mathcal{S}}|\mu_{\mathcal{S}} + K_{\mathcal{S}\mathcal{S}}\Gamma_{\mathcal{S}\mathcal{S}}^{-1}V_{\mathcal{S}\mathcal{D}}, K_{\mathcal{S}\mathcal{S}}\Gamma_{\mathcal{S}\mathcal{S}}^{-1}K_{\mathcal{S}\mathcal{S}})$ where $\Gamma_{\mathcal{S}\mathcal{S}} \triangleq K_{\mathcal{S}\mathcal{S}} + K_{\mathcal{S}\mathcal{D}}R_{\mathcal{D}\mathcal{D}}^{-1}K_{\mathcal{D}\mathcal{S}}$ and $V_{\mathcal{S}\mathcal{D}} = K_{\mathcal{S}\mathcal{D}}R_{\mathcal{D}\mathcal{D}}^{-1}(y_{\mathcal{D}} - \mu_{\mathcal{D}})$ such that $R_{\mathcal{D}\mathcal{D}} \triangleq K_{\mathcal{D}\mathcal{D}} - Q_{\mathcal{D}\mathcal{D}} + \sigma_n^2 I$. But, computing $p(f_{\mathcal{S}}|y_{\mathcal{D}})$ generally incurs $\mathcal{O}(|\mathcal{D}|^3)$ time. To reduce to linear time and achieve further scalability via parallelization, the work of Low et al. (2015) has proposed LMA by approximating $R_{\mathcal{D}\mathcal{D}}$ with $S_{\mathcal{D}\mathcal{D}}$ (1) based on the covariance function in (9), which interestingly induces the same $q(f_{\mathcal{S}})$ and predictive distribution (Appendix I) as that produced by our variational inference framework (see, respectively, (6) in Section 3 and (48) in Appendix F, the latter of which relies on a specific choice of approximation $q(f_{\mathcal{U}_i}|f_{\mathcal{S}}, y_{\mathcal{D}})$ (47) to the test conditional $p(f_{\mathcal{U}_i}|f_{\mathcal{S}}, y_{\mathcal{D}})$). So, LMA is unified with our framework.

4.3. Key insight: Different variational SGPR models evolve from varying noise correlation structures

It is well-known that, among existing SGPR models, DTC and FITC are deemed variationally optimal as they minimize the KL distance to a FGPR model with independently distributed noises (i.e., $S_{\mathcal{D}\mathcal{D}} = \sigma_n^2 I$ (Titsias, 2009a) or $S_{\mathcal{D}\mathcal{D}} = \mathrm{diag}[K_{\mathcal{D}\mathcal{D}} - Q_{\mathcal{D}\mathcal{D}}] + \sigma_n^2 I$ (Titsias, 2009b)). However, empirical results in the existing literature show that FITC and DTC often predict more poorly than PIC (Hoang et al., 2015), and LMA (Low et al., 2015)[8], which are usually explained by empirically inspecting how well these SGPR models approximate the prior covariance matrix of the FGPR model (Low et al., 2015; Quiñonero-Candela & Rasmussen, 2005; Snelson, 2007). But, to the best of our knowledge, it has not been explicitly explained why SGPR models like PIC and LMA, despite being able to provide more refined approximations of the prior covariance matrix of the FGPR model, do not minimize their KL distance to the FGPR model.

Our unification results (Sections 4.1 and 4.2) have explained this by giving a theoretical justification based on the correlation structure of the noise process model: Specifically, FITC and DTC can only minimize their KL distances to a FGPR model under the assumption of independently distributed noises, which is often violated in many real-world datasets where observation noises tend to be correlated (Huizenga & Molenaar, 1995; Koochakzadeh et al., 2015; Rasmussen & Williams, 2006). However, they do not necessarily minimize their KL distances to a FGPR model in such cases. In light of this fact, our unification results reveal that different SGPR models minimize their KL distances to a FGPR model under varying assumptions of correlation structure of the noise process model: For example, Section 4.1 (4.2) shows that when our variational inference framework minimizes the KL distance to a FGPR model (Section 3) under the assumption of $B = 0$ ($B > 0$) and the covariance function $k^{\epsilon}_{\mathbf{x}\mathbf{x}'}$ in (9) for the noise process model, it recovers the same $q(f_{\mathcal{S}})$ and predictive distribution $q(f_{\mathcal{U}_i}|y_{\mathcal{D}})$ induced by PIC (LMA).

5. Distributed Variational Inference for Hyperparameter Learning

Recall from Section 3 that plugging the derived $q(f_{\mathcal{S}})$ (6) into the variational lower bound $\mathcal{L}(q, \mathcal{Z})$ (4) gives

$$\begin{aligned}&\mathcal{R}(\mathcal{Z}) = \max_q \mathcal{L}(q, \mathcal{Z})\\ &= \log \mathcal{N}(y_{\mathcal{D}}|\mu_{\mathcal{D}}, S_{\mathcal{D}\mathcal{D}} + Q_{\mathcal{D}\mathcal{D}}) - 0.5\mathrm{Tr}[S_{\mathcal{D}\mathcal{D}}^{-1}(K_{\mathcal{D}\mathcal{D}} - Q_{\mathcal{D}\mathcal{D}})]\end{aligned} \tag{10}$$

where the last equality is derived in Appendix J. Hyperparameter learning then involves iteratively refining the estimate of hyperparameters $\mathcal{Z}$ via gradient ascent to improve the value of $\mathcal{R}(\mathcal{Z})$. Its time complexity depends on the efficiency of computing the gradient $\partial\mathcal{R}/\partial\mathcal{Z}$ per iteration of gradient ascent. As shown in Appendix K, differentiating both sides of (10) with respect to $z \in \mathcal{Z}$ yields

$$\begin{aligned}&\partial\mathcal{R}/\partial z =\\ &-\frac{1}{2}\mathrm{Tr}\left[\frac{\partial P_{\mathcal{D}\mathcal{D}}}{\partial z}W_{\mathcal{D}\mathcal{D}} + P_{\mathcal{D}\mathcal{D}}\frac{\partial W_{\mathcal{D}\mathcal{D}}}{\partial z} - S_{\mathcal{D}\mathcal{D}}\frac{\partial P_{\mathcal{D}\mathcal{D}}}{\partial z}\right]\\ &-\frac{1}{2}\mathrm{Tr}\left[\Gamma_{\mathcal{S}\mathcal{S}}^{-1}\frac{\partial \Gamma_{\mathcal{S}\mathcal{S}}}{\partial z} - K_{\mathcal{S}\mathcal{S}}^{-1}\frac{\partial K_{\mathcal{S}\mathcal{S}}}{\partial z}\right]\\ &+\frac{1}{2}\mathrm{Tr}\left[V_{\mathcal{S}\mathcal{D}}^{\top}\frac{\partial \Gamma_{\mathcal{S}\mathcal{S}}^{-1}}{\partial z}V_{\mathcal{S}\mathcal{D}} + 2\frac{\partial V_{\mathcal{S}\mathcal{D}}^{\top}}{\partial z}\Gamma_{\mathcal{S}\mathcal{S}}^{-1}V_{\mathcal{S}\mathcal{D}}\right]\end{aligned} \tag{11}$$

where $W_{\mathcal{D}\mathcal{D}} \triangleq K_{\mathcal{D}\mathcal{D}} - Q_{\mathcal{D}\mathcal{D}} + (y_{\mathcal{D}} - \mu_{\mathcal{D}})(y_{\mathcal{D}} - \mu_{\mathcal{D}})^{\top}$. However, computing $\partial\mathcal{R}/\partial z$ directly using (11) is prohibitively expensive as it involves computing the matrix derivative $\partial P_{\mathcal{D}\mathcal{D}}/\partial z = -P_{\mathcal{D}\mathcal{D}}\,(\partial S_{\mathcal{D}\mathcal{D}}/\partial z)\,P_{\mathcal{D}\mathcal{D}}$ which

[8] In (Low et al., 2015), LMA outperforms PIC on the same dataset used by Hoang et al. (2015) to compare PIC and DTC.

generally incurs $\mathcal{O}\left(|\mathcal{D}|^3\right)$ time. In the rest of this section, we will exploit the B-block-banded structure of $P_{\mathcal{D}\mathcal{D}}$ (Section 2) for deriving an efficient reformulation of $\partial\mathcal{R}/\partial z$ (11) that can be computed in linear time in the data size $|\mathcal{D}|$. More importantly, it is amenable to parallelization on distributed machines/cores by constructing and communicating summaries of data clusters to be described next:

Definition 1 (B-th order Markov Cluster) *A B-th order Markov cluster k is defined as $\mathcal{M}_k^B \triangleq \bigcup_{i\in\mathcal{B}^+(k)} \mathcal{D}_i \supset \mathcal{D}_k^B$ for $k = 1,\ldots,M$ where $\mathcal{B}^+(k)$ and $\mathcal{D}_k^B$ are previously defined in Theorem 1 and just below (1), respectively.*

From Definition 1, $\mathcal{D} = \bigcup_{k=1}^M \mathcal{M}_k^B$ and $\mathcal{M}_k^B \cap \mathcal{M}_{k'}^B \neq \emptyset$ if $|k-k'| \leq B$ and $B > 0$. Each B-th order Markov cluster k of size $|\mathcal{M}_k^B| = \mathcal{O}\left(B|\mathcal{D}|/M\right) = \mathcal{O}(B|\mathcal{S}|)$[6] is assigned to a separate machine/core k which constructs its local summary, as defined below:

Definition 2 (Local Summary) *The local summary of a B-th order Markov cluster k is defined as a tuple $\mathcal{P}_k \triangleq (\alpha_k, \beta_k, \{\alpha_k^z, \beta_k^z, \delta_k^z, \psi_k^z\}_{z\in\mathcal{Z}})$ where α_k and β_k are previously defined in Theorem 1, $\alpha_k^z \triangleq \partial\alpha_k/\partial z$, $\beta_k^z \triangleq \partial\beta_k/\partial z$,*

$$\delta_k^z \triangleq \sum_{i,j\in\mathcal{B}^+(k)} \mathrm{Tr}\left[\frac{\partial}{\partial z}\Big(G^k_{\mathcal{D}_i\mathcal{D}_k} G^{k^{-1}}_{\mathcal{D}_k\mathcal{D}_k} G^k_{\mathcal{D}_k\mathcal{D}_j} W_{\mathcal{D}_j\mathcal{D}_i}\Big)\right],$$

$$\psi_k^z \triangleq \sum_{i,j\in\mathcal{B}^+(k)} \mathrm{Tr}\left[\frac{\partial}{\partial z}\Big(G^k_{\mathcal{D}_i\mathcal{D}_k} G^{k^{-1}}_{\mathcal{D}_k\mathcal{D}_k} G^k_{\mathcal{D}_k\mathcal{D}_j}\Big) S_{\mathcal{D}_j\mathcal{D}_i}\right].$$

The local summaries $\mathcal{P}_1,\ldots,\mathcal{P}_M$ can be computed independently and hence in parallel on M distributed machines/cores: To construct local summary $\mathcal{P}_k$, each machine/core k only needs access to its Markov cluster $\mathcal{M}_k^B$, the corresponding noisy outputs $y_{\mathcal{M}_k^B}$, and a common set $\mathcal{S}$ of inducing inputs known to all machines/cores. Evaluating the derivative terms in local summary $\mathcal{P}_k$ essentially requires computing terms such as $\partial G^k_{\mathcal{D}_i\mathcal{D}_k}/\partial z$ for $i \in \mathcal{B}^+(k)$. To compute the latter terms, we exploit the result of Lemma 1 for computing $\partial G^k/\partial z$ in terms of $\partial S_{\mathcal{D}_k^B\mathcal{D}_k^B}/\partial z$, $\partial S_{\mathcal{D}_k^B\mathcal{D}_k}/\partial z$, and $\partial S_{\mathcal{D}_k\mathcal{D}_k}/\partial z$. For each Markov cluster $\mathcal{M}_k^B$, computing G^k (and hence $\partial G^k/\partial z$) incurs only $\mathcal{O}(B^3|\mathcal{S}|^3)$ time, as discussed in the paragraph after Lemma 1. Its corresponding local summary $\mathcal{P}_k$ can then be computed in $\mathcal{O}(B^2|\mathcal{S}|^3)$ time. So, the overall time complexity of computing local summary $\mathcal{P}_k$ of Markov cluster $\mathcal{M}_k^B$ in each parallel machine/core k is $\mathcal{O}(B^3|\mathcal{S}|^3)$ which is independent of the data size $|\mathcal{D}|$. Alternatively, $\mathcal{P}_1,\ldots,\mathcal{P}_M$ can be constructed sequentially in $\mathcal{O}(|\mathcal{D}||\mathcal{S}|^2B^3) = \mathcal{O}(MB^3|\mathcal{S}|^3)$ time that is linear in $|\mathcal{D}|$. Every machine/core k then communicates the local summary $\mathcal{P}_k$ of its assigned Markov cluster $\mathcal{M}_k^B$ to a central machine that will assimilate these local summaries into a global summary defined as follows:

Definition 3 (Global Summary) *The global summary is defined as a tuple $\mathcal{P}_* \triangleq (\alpha, \beta, \{\alpha^z, \beta^z, \phi^z\}_{z\in\mathcal{Z}})$ where*

$$\alpha \triangleq \sum_{k=1}^M \alpha_k\,,\quad \beta \triangleq K_{\mathcal{S}\mathcal{S}} + \sum_{k=1}^M \beta_k\,,\quad \alpha^z \triangleq \sum_{k=1}^M \alpha_k^z\,,$$
$$\beta^z \triangleq \frac{\partial K_{\mathcal{S}\mathcal{S}}}{\partial z} + \sum_{k=1}^M \beta_k^z\,,\quad \textit{and}\quad \phi^z \triangleq \sum_{k=1}^M \delta_k^z - \psi_k^z\,.$$

Our main result to follow presents an efficient reformulation of $\partial\mathcal{R}/\partial z$ (11) by exploiting the global summary $\mathcal{P}_*$:

Theorem 2 *$\partial\mathcal{R}/\partial z$ (11) can be re-expressed in terms of global summary $\mathcal{P}_* = (\alpha, \beta, \{\alpha^z, \beta^z, \phi^z\}_{z\in\mathcal{Z}})$ as follows:*

$$\frac{\partial\mathcal{R}}{\partial z} = \frac{1}{2}\mathrm{Tr}\left[K_{\mathcal{S}\mathcal{S}}^{-1}\frac{\partial K_{\mathcal{S}\mathcal{S}}}{\partial z}\right] + \alpha^{z\top}\beta^{-1}\alpha - \frac{1}{2}\mathrm{Tr}\left[\beta^{-1}\beta^z\right] - \frac{1}{2}\alpha^\top\beta^{-1}\beta^z\beta^{-1}\alpha - \frac{1}{2}\phi^z. \tag{12}$$

Its proof is in Appendix L. Since α and α^z are column vectors of size $|\mathcal{S}|$, β and β^z are matrices of size $|\mathcal{S}|$ by $|\mathcal{S}|$, and ϕ^z is a scalar, computing $\partial\mathcal{R}/\partial z$ using (12) only incurs $\mathcal{O}(|\mathcal{S}|^3)$ time given the global summary $\mathcal{P}_*$. So, the overall time complexity of computing $\partial\mathcal{R}/\partial z$ (12) in a distributed manner on M parallel machines/cores (sequentially) is still $\mathcal{O}(B^3|\mathcal{S}|^3)$ ($\mathcal{O}(|\mathcal{D}||\mathcal{S}|^2B^3) = \mathcal{O}(MB^3|\mathcal{S}|^3)$).

6. Experiments and Discussion

This section empirically evaluates the predictive performance and scalability of distributed variational SGPR models unified by our framework (Section 5) such as the distributed variants of PIC and LMA and the current state-of-the-art distributed variant of DTC (Gal et al., 2014), which we respectively call dPIC, dLMA, and dDTC[9], on two real-world datasets:

(a) The AIMPEAK dataset (Chen et al., 2013a) contains 41850 observations of traffic speeds (km/h) along 775 road segments of an urban road network during morning peak hours on April 20, 2011. Each observation comprises a 5-dimensional input vector featuring the length, number of lanes, speed limit, direction of the road segment as well as its recording time which is discretized into 54 five-minute time slots. The outputs correspond to the traffic speeds.

(b) The AIRLINE dataset (Hensman et al., 2013; Hoang et al., 2015) contains 2055733 information records of commercial flights in the USA from January to April 2008. The input denotes a 8-dimensional feature vector of the age of the aircraft (year), travel distance (km), airtime, departure and arrival time (min), day of the week, day of the month, and month. The output is the delay time (min) of the flight.

Both datasets are modeled using GP with correlated noises (Section 2) whose prior covariance matrix is defined using the squared exponential covariance function described in

[9]The induced variational lower bound of dDTC (Eq. (10)) is equivalent to that of the Dist-VGP framework of Gal et al. (2014).

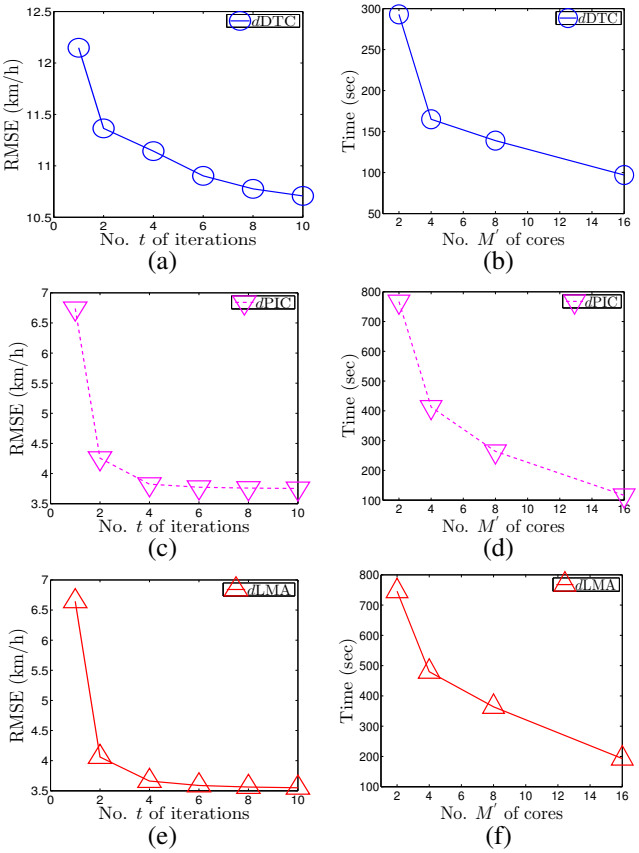

Figure 1. Graphs of RMSEs achieved by (a) dDTC, (c) dPIC, and (e) dLMA vs. number t of iterations, and graphs of the total training times incurred by (b) dDTC, (d) dPIC, and (f) dLMA vs. number M' of parallel cores for the AIMPEAK dataset.

Section 2. For dPIC and dLMA models, the prior covariance of the noise process is constructed using (1) (respectively, with Markov order $B = 0$ and 1) and the covariance function in (9). Likewise, for dDTC, the prior covariance of the noise process is constructed using (1) with $B = 0$ and covariance function $k^{\epsilon}_{\mathbf{x}\mathbf{x}'} \triangleq \sigma_n^2 I$. Both the training and test data are then partitioned evenly into M blocks using k-means (i.e., $k = M$). All experiments are run on a Linux system with Intel® Xeon® E5-2670 at 2.6GHz with 96 GB memory and 32 processing cores. Our distributed variational SGPRs are implemented using Armadillo linear algebra library for C++ (Sanderson, 2010). For each tested model, we report the (a) *root mean square error* (RMSE), $\sqrt{|\mathcal{U}|^{-1}\sum_{\mathbf{x}\in\mathcal{U}}(y_{\mathbf{x}} - \widehat{y}_{\mathbf{x}})^2}$, of its predictions $\{\widehat{y}_{\mathbf{x}}\}_{\mathbf{x}\in\mathcal{U}}$, (b) training time vs. no. of iterations, and (c) training time vs. no. of parallel cores. The results for each model are evaluated on 5% of the dataset with respect to its best configuration (e.g., learning rate, no. of inducing inputs, etc.).

AIMPEAK Dataset. We randomly remove 50 data points from the original dataset so that the experimented data can be partitioned evenly into $M = 100$ blocks. The empirical results, observations, and analysis are described below:

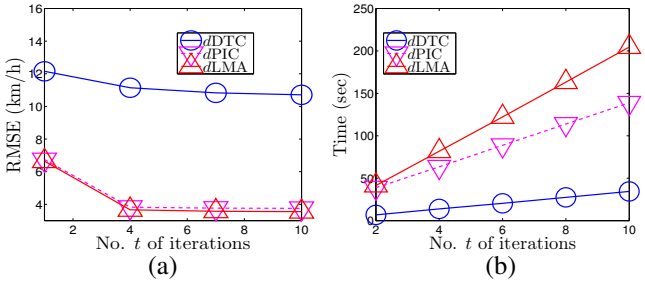

Figure 2. Graphs of (a) RMSEs and (b) total incurred times of dDTC, dPIC, and dLMA vs. number t of iterations with $M' = 16$ computing cores and $M = 100$ blocks for AIMPEAK dataset.

(a) Figs. 1 and 2 show results of RMSEs, incurred times, and parallel efficiencies of dPIC, dDTC, and dLMA averaged over 5 random instances with varying number t of iterations. In particular, it can be observed from both Figs. 1 and 2 that the RMSEs of all tested methods decrease rapidly (see Figs 1a, 1c, and 1e) while their total incurred training times only increase linearly in the number t of iterations (see Fig. 2b), which shows the effectiveness of our distributed hyperparameter learning framework (Section 5) for these variational SGPR models;

(b) Fig. 1 also shows results of the total training times incurred by dPIC, dLMA, and dDTC over 10 iterations with varying number $M' = 2, 4, 8, 16$ of parallel cores. As expected, it can be observed from Figs. 1b, 1d, and 1f that the training times of the tested models decrease gradually when the number of computing cores increases, which corroborates our complexity analysis of distributed SGPR models in Section 5. More specifically, our experiment reveals that using $M' = 16$ cores, dPIC (116 seconds), dLMA (193 seconds), and dDTC (97 seconds) can achieve speedups (i.e., parallel efficiencies) of 5.5 to 7.2 over their centralized counterparts (i.e., $M' = 1$): PIC (836 seconds), LMA (1266 seconds), and DTC (537 seconds).

(c) Fig. 2b further shows that dDTC consistently incurs less training time than both dPIC and dLMA with varying number t of learning iterations. This is expected because the primary cost of computing a local summary for dDTC, which involves computing G^k in Lemma 1, is constant[10] while that of dPIC and dLMA grows cubically in the block size $|\mathcal{D}|/M$. On the other hand, Fig. 2a, however, reveals that the RMSEs achieved by dPIC (3.75) and dLMA (3.54) are both significantly lower than that achieved by dDTC (10.70). This inferior performance of dDTC is also expected because of its more restrictive assumption of deterministic relation between the training and inducing outputs (Hoang et al., 2015).

AIRLINE Dataset. We extract 2M data points from the original dataset for experiment to guarantee an even par-

[10]For dDTC, since $S_{\mathcal{D}\mathcal{D}} = \sigma_n^2 I$, it follows straightforwardly that $G^k = \sigma_n^{-2} I$ which can be constructed in constant time.

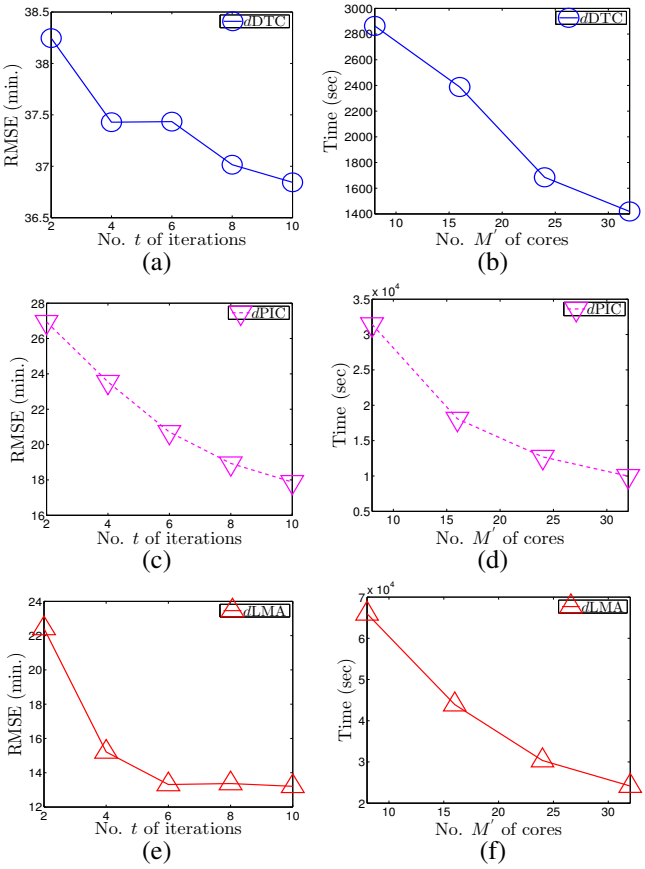

Figure 3. Graphs of RMSEs achieved by (a) dDTC, (c) dPIC, and (e) dLMA vs. number t of iterations, and graphs of the total training times incurred by (b) dDTC, (d) dPIC, and (f) dLMA vs. number M' of parallel cores for the AIRLINE dataset.

tition into $M = 2000$ blocks. Figs. 3 and 4 show results of RMSEs, incurred time, and parallel efficiencies of dPIC, dDTC, and dLMA averaged over 5 random instances with varying number t of iterations. The observations are mostly similar to that of the AIMPEAK dataset: From Figs. 3a, 3c, and 3e, the RMSEs of dPIC, dDTC, and dLMA decrease quickly while their total incurred training times only increase linearly in the number t of iterations (Fig. 4b). Figs. 3b, 3d, and 3f then show a rapid decrease of their total training times with increasing number of processing cores. Our experiments reveal that using $M' = 32$ cores, dLMA (24151 seconds), dPIC (9981 seconds), and dDTC (1419 seconds) incur less than 6.7 hours to optimize their hyperparameters and can achieve significant speedups from 12.69 to 13.58 over their centralized counterparts: LMA (306476 seconds), PIC (135542 seconds), and DTC (19426 seconds). It can also be observed from Fig. 4b that dDTC incurs less training time than dPIC and dLMA but, as observed from Fig. 4a, both dLMA (13.20) and dPIC (17.88) outperform dDTC (36.84) by a huge margin.

Finally, the predictive performance of our proposed distributed hyperparameter learning frameworks, dLMA and

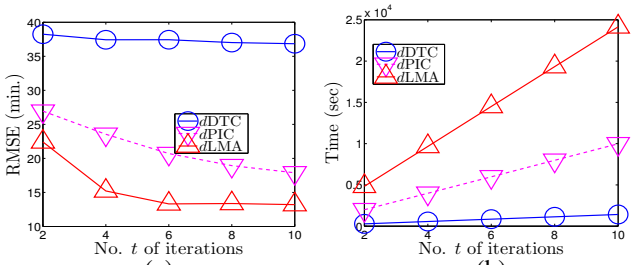

Figure 4. Graphs of (a) RMSEs and (b) total incurred times of dDTC, dPIC, and dLMA vs. number t of iterations with $M' = 32$ computing cores and $M = 2000$ blocks for AIRLINE dataset.

dPIC, are further evaluated on two benchmark settings of the AIRLINE dataset to compare with the previously best reported RMSE results of the existing state-of-the-art distributed methods such as Dist-VGP (Gal et al., 2014) and rBCM (Deisenroth & Ng, 2015). All results are reported in Table 1 below, which essentially shows that dPIC and dLMA significantly outperform Dist-VGP and rBCM in both settings. Notably, dLMA (16.50 and 13.20) manages to reduce the previously best reported RMSEs (27.10 and 34.40) by 39.11% and 61.62%, respectively.

	Dist-VGP	rBCM	dPIC	dLMA
700K/100K	33.00	27.10	**21.09**	**16.50**
2M/100K	35.30	34.40	**17.88**	**13.20**

Table 1. RMSEs achieved by existing distributed/parallel methods on two standard benchmark settings (training/test data sizes) of AIRLINE dataset: (a) 700K/100K and (b) 2M/100K. The results (RMSEs) of Dist-VGP (Gal et al., 2014) and rBCM (Deisenroth & Ng, 2015) are reported from their respective papers.

7. Conclusion

This paper describes a novel distributed variational inference framework that unifies many parallel SGPR models (e.g., DTC, FITC, FIC, PITC, PIC, LMA) for scalable hyperparameter learning, consequently reducing their incurred linear time per iteration of gradient ascent significantly. To achieve this, our framework exploits a structure of correlated noise process model that represents the observation noises as a finite realization of a B-th order Gaussian Markov random process. By varying the Markov order and covariance function for the noise process model, different variational SGPR models result. This consequently allows the correlation structure of the noise process model to be characterized for which a particular variational SGPR model is optimal; in other words, different SGPR models minimize their KL distance to a FGPR model under varying characterizations of the noise correlation structure. Empirical evaluation on two real-world datasets show that our proposed framework can achieve significantly better predictive performance than the state-of-the-art distributed variational DTC (Gal et al., 2014) and distributed GPs (Deisenroth & Ng, 2015) while preserving scalability to big data.

Acknowledgments. This research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its International Research Centre in Singapore Funding Initiative.

References

Asif, A. and Moura, J. M. F. Block matrices with L-block-banded inverse: Inversion algorithms. *IEEE Trans. Signal Processing*, 53(2):630–642, 2005.

Campbell, T., Straub, J., Fisher III, J. W., and How, J. P. Streaming, distributed variational inference for Bayesian nonparametrics. In *Proc. NIPS*, pp. 280–288, 2015.

Cao, N., Low, K. H., and Dolan, J. M. Multi-robot informative path planning for active sensing of environmental phenomena: A tale of two algorithms. In *Proc. AAMAS*, pp. 7–14, 2013.

Chen, J., Low, K. H., Tan, C. K.-Y., Oran, A., Jaillet, P., Dolan, J. M., and Sukhatme, G. S. Decentralized data fusion and active sensing with mobile sensors for modeling and predicting spatiotemporal traffic phenomena. In *Proc. UAI*, pp. 163–173, 2012.

Chen, J., Cao, N., Low, K. H., Ouyang, R., Tan, C. K.-Y., and Jaillet, P. Parallel Gaussian process regression with low-rank covariance matrix approximations. In *Proc. UAI*, pp. 152–161, 2013a.

Chen, J., Low, K. H., and Tan, C. K.-Y. Gaussian process-based decentralized data fusion and active sensing for mobility-on-demand system. In *Proc. RSS*, 2013b.

Chen, J., Low, K. H., Jaillet, P., and Yao, Y. Gaussian process decentralized data fusion and active sensing for spatiotemporal traffic modeling and prediction in mobility-on-demand systems. *IEEE Transactions on Automation Science and Engineering*, 12(3):901–921, 2015.

Deisenroth, M. P. and Ng, J. W. Distributed Gaussian processes. In *Proc. ICML*, 2015.

Dolan, J. M., Podnar, G., Stancliff, S., Low, K. H., Elfes, A., Higinbotham, J., Hosler, J. C., Moisan, T. A., and Moisan, J. Cooperative aquatic sensing using the telesupervised adaptive ocean sensor fleet. In *Proc. SPIE Conference on Remote Sensing of the Ocean, Sea Ice, and Large Water Regions*, volume 7473, 2009.

Gal, Y., van der Wilk, M., and Rasmussen, C. Distributed variational inference in sparse gaussian process regression and latent variable models. In *Proc. NIPS*, pp. 3257–3265, 2014.

Goldberg, P. W., William, C. K. I., and Bishop, C. M. Regression with input-dependent noise: A Gaussian process treatment. In *Proc. NIPS*, pp. 493–499, 1997.

Hensman, J., Fusi, N., and Lawrence, N. D. Gaussian processes for big data. In *Proc. UAI*, pp. 282–290, 2013.

Hoang, T. N., Low, K. H., Jaillet, P., and Kankanhalli, M. Active learning is planning: Nonmyopic ϵ-Bayes-optimal active learning of Gaussian processes. In *Proc. ECML/PKDD Nectar Track*, pp. 494–498, 2014a.

Hoang, T. N., Low, K. H., Jaillet, P., and Kankanhalli, M. Nonmyopic ϵ-Bayes-Optimal Active Learning of Gaussian Processes. In *Proc. ICML*, pp. 739–747, 2014b.

Hoang, T. N., Hoang, Q. M., and Low, K. H. A unifying framework of anytime sparse Gaussian process regression models with stochastic variational inference for big data. In *Proc. ICML*, pp. 569–578, 2015.

Huizenga, H. M. and Molenaar, P. C. M. Equivalent source estimation of scalp potential fields contaminated by heteroscedastic and correlated noise. *Brain Topography*, 8 (1):13–33, 1995.

Kersting, K., Plagemann, C., Pfaff, P., and Burgard, W. Most likely heteroscedastic Gaussian process regression. In *Proc. ICML*, pp. 393–400, 2007.

Koochakzadeh, A., Malek-Mohammadi, M., Babaie-Zadeh, M., and Skoglund, M. Multi-antenna assisted spectrum sensing in spatially correlated noise environments. *Signal Processing*, 108:69–76, 2015.

Lázaro-Gredilla, M. and Titsias, M. K. Variational heteroscedastic Gaussian process regression. In *Proc. ICML*, pp. 841–848, 2011.

Lázaro-Gredilla, M., Quiñonero-Candela, J., Rasmussen, C. E., and Figueiras-Vidal, A. R. Sparse spectrum Gaussian process regression. *Journal of Machine Learning Research*, pp. 1865–1881, 2010.

Ling, C. K., Low, K. H., and Jaillet, P. Gaussian process planning with Lipschitz continuous reward functions: Towards unifying Bayesian optimization, active learning, and beyond. In *Proc. AAAI*, pp. 1860–1866, 2016.

Low, K. H., Dolan, J. M., and Khosla, P. Adaptive multi-robot wide-area exploration and mapping. In *Proc. AAMAS*, pp. 23–30, 2008.

Low, K. H., Dolan, J. M., and Khosla, P. Information-theoretic approach to efficient adaptive path planning for mobile robotic environmental sensing. In *Proc. ICAPS*, pp. 233–240, 2009.

Low, K. H., Dolan, J. M., and Khosla, P. Active Markov information-theoretic path planning for robotic environmental sensing. In *Proc. AAMAS*, pp. 753–760, 2011.

Low, K. H., Chen, J., Dolan, J. M., Chien, S., and Thompson, D. R. Decentralized active robotic exploration and mapping for probabilistic field classification in environmental sensing. In *Proc. AAMAS*, pp. 105–112, 2012.

Low, K. H., Chen, J., Hoang, T. N., Xu, N., and Jaillet, P. Recent advances in scaling up Gaussian process predictive models for large spatiotemporal data. In *Proc. DyDESS*, 2014a.

Low, K. H., Xu, N., Chen, J., Lim, K. K., and Özgül, E. B. Generalized online sparse Gaussian processes with application to persistent mobile robot localization. In *Proc. ECML/PKDD Nectar Track*, pp. 499–503, 2014b.

Low, K. H., Yu, J., Chen, J., and Jaillet, P. Parallel Gaussian process regression for big data: Low-rank representation meets Markov approximation. In *Proc. AAAI*, pp. 2821–2827, 2015.

Murray-Smith, R. and Girard, A. Gaussian process priors with ARMA noise models. In *Proc. Irish Signals Systems Conference*, pp. 147–153, 2001.

Ouyang, R., Low, K. H., Chen, J., and Jaillet, P. Multi-robot active sensing of non-stationary Gaussian process-based environmental phenomena. In *Proc. AAMAS*, pp. 573–580, 2014.

Podnar, G., Dolan, J. M., Low, K. H., and Elfes, A. Telesupervised remote surface water quality sensing. In *Proc. IEEE Aerospace Conference*, 2010.

Quiñonero-Candela, J. and Rasmussen, C. E. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Sanderson, Conrad. Armadillo: An open source c++ linear algebra library for fast prototyping and computationally intensive experiments. Technical report, NICTA, 2010.

Schwaighofer, A. and Tresp, V. Transductive and inductive methods for approximate Gaussian process regression. In *Proc. NIPS*, pp. 953–960, 2003.

Seeger, M., Williams, C. K. I., and Lawrence, N. D. Fast forward selection to speed up sparse Gaussian process regression. In *Proc. AISTATS*, 2003.

Smola, A. J. and Bartlett, P. L. Sparse greedy Gaussian process regression. In *Proc. NIPS*, pp. 619–625, 2001.

Snelson, E. and Gharahmani, Z. Sparse Gaussian processes using pseudo-inputs. In *Proc. NIPS*, pp. 1259–1266, 2005.

Snelson, E. L. *Flexible and efficient Gaussian process models for machine learning*. Ph.D. Thesis, University College London, London, UK, 2007.

Snelson, E. L. and Ghahramani, Z. Local and global sparse Gaussian process approximation. In *Proc. AISTATS*, 2007.

Titsias, M. K. Variational learning of inducing variables in sparse Gaussian processes. In *Proc. AISTATS*, pp. 567–574, 2009a.

Titsias, M. K. Variational model selection for sparse Gaussian process regression. Technical report, School of Computer Science, University of Manchester, 2009b.

Wainwright, M. J. and Jordan, M. I. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

Xu, N., Low, K. H., Chen, J., Lim, K. K., and Özgül, E. B. GP-Localize: Persistent mobile robot localization using online sparse Gaussian process observation model. In *Proc. AAAI*, pp. 2585–2592, 2014.

Yu, J., Low, K. H., Oran, A., and Jaillet, P. Hierarchical Bayesian nonparametric approach to modeling and learning the wisdom of crowds of urban traffic route planning agents. In *Proc. IAT*, pp. 478–485, 2012.

Zhang, Y., Hoang, T. N., Low, K. H., and Kankanhalli, M. Near-optimal active learning of multi-output Gaussian processes. In *Proc. AAAI*, pp. 2351–2357, 2016.